

ربط قواعد البيانات Access و Sql server بلغة الـ #c باستخدام تقنية ado.net



hamza


إهداء
إلى غاليتي الحبيبة
عشقي الأزلي
إن لم أقل حبي السرمدي
الهبة الثمينة التي وهبني الله إياها
أمل حياتي
نور عيوني
شمسي .. قمري ..
فرحة عمري
نبع الحنان الذي لا ينضب
قلب العطاء الدائم الذي لا يتوقف
ولا ينتظر شكرا أو عرفانا
إليك يا حبيبتي الغالية
امي
ست الحبايب

نتناول في هذا الكتاب شرح ربط برامج السي شارب بقواعد البيانات الأكسس والإس كيو إل سرفر وحتى وان كان اغلب الشرح هو بالإكسس ولكن ركزوا على هذه النقطة أن التغيير البسيط في الشرح مابين القاعدتين هو اختلاف مجال الأسماء واختلاف نص الاتصال و **تستطيع استخدام الشرح المقدم في هذا الكتاب في لغة الاس كيو إل سرفر فما عليك بعد النقطتين السابقتين إلا أن تغير اشتقاق الكائنات التي تتبع الـ (OLDEB) مثلاً فئة (OleDbConnection) هي فئة الإتصال بإستخدام قاعدة بيانات من نوع الأكسس والأختلاف انه عند استخدام قاعدة بيانات الإس كيو إل سرفر اننا نغير كلمة (OLDEB) بكلمة (SQL) يعني سيصبح هكذا (SQLConnection) وفي كل الفئات الذي تبدأ بكلمة (OLDEB) ما علينا سوى إستبدالها بكلمة(SQL) .**

وإن ما دعاني لأكتب هذا الكتاب هو ما لاقيته من صعوبة في وجود مناهج بالعربية تقدم الشرح المبسط لهذه الإجراءات وأرجوا أن يستفيد من ما كتبته في هذا الكتاب أي شخص وأعذروني عن أي تقصير وأن كان هناك أية أسئلة ارجوا منكم أن تراسلوني على احد الإيميلات .

امتناني لكم

حمزة علي محمد عبده حزام البُعسي

برمجة حاسوب - اليمن

https://www.facebook.com/hamzawey2035

@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

سنبدأ أول درس من دروسنا البسيطة بدرس كيفية إنشاء قاعدة بيانات بسيطة في الأكسس .

نضغط على قائمة ابدأ ثم كافة البرامج ثم Microsoft Office ثم Microsoft Office Access وبعد أن تفتح لنا واجهة البرنامج نضغط على (قاعدة بيانات فارغة) كما في الصورة



1 / قاعدة بيانات فارغة جديدة

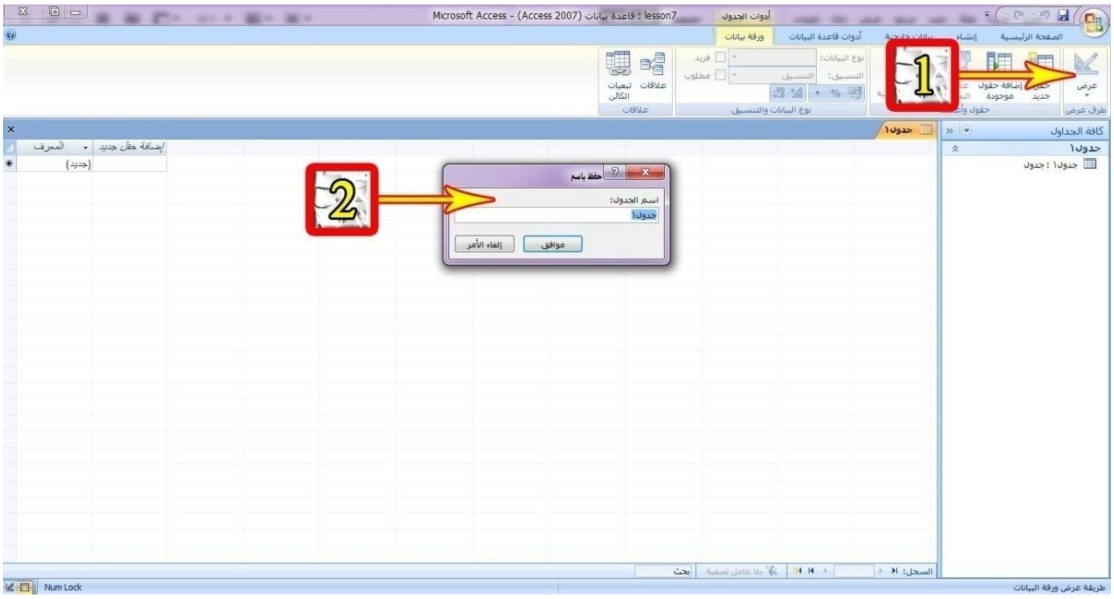
2 / تسمية قاعدة البيانات بالاسم الذي تشاء

3 / مكان حفظ قاعدة البيانات التي سوف تنشئها اضغط على استعراض واختر المكان الذي تريده

وفي هذا المثال أنا استخدمت الاسم lesson7
ومن ثم الضغط على إنشاء ليتم إنشاء قاعدة البيانات الخاصة بك .

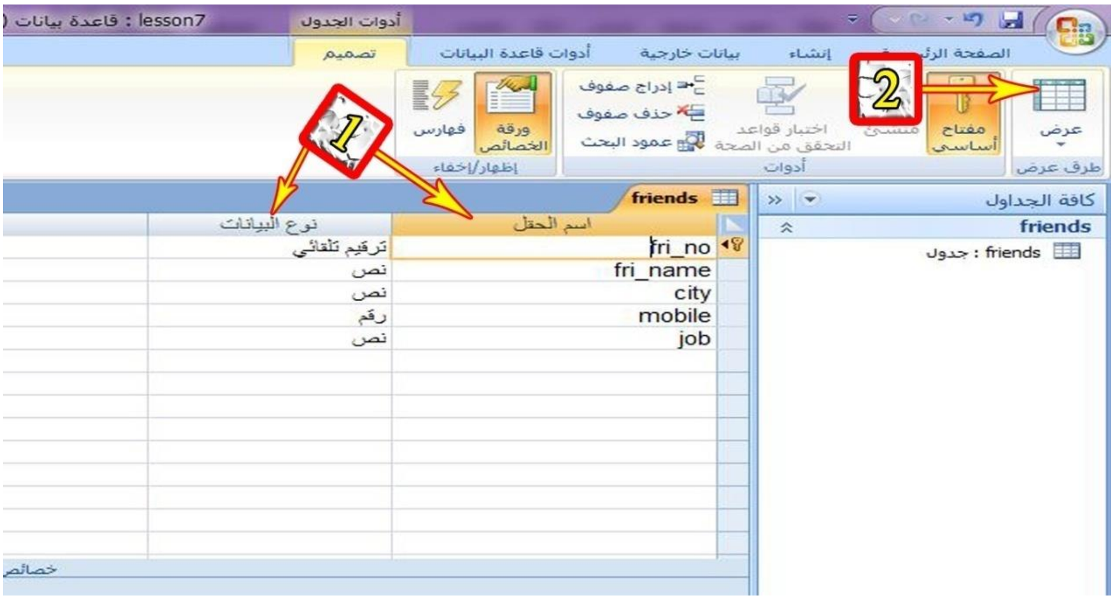
- تفتح لنا هذه الواجهة

https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com



1 / الضغط على (عرض) في أعلى يمين الشاشة لإظهار تصميم الجدول الجديد
2 / يطلب منا أن نقوم بتسمية الجدول الجديد اختر الاسم الذي تريده .

و أنا قد أسميت الجدول الذي سوف نقوم بالتطبيق عليه بالاسم friends
بعد ذلك تفتح لنا صفحة التصميم فنقوم بتصميم جدولنا كما نريد وكما نحتاج من أعمدة وأنواعها البيانية

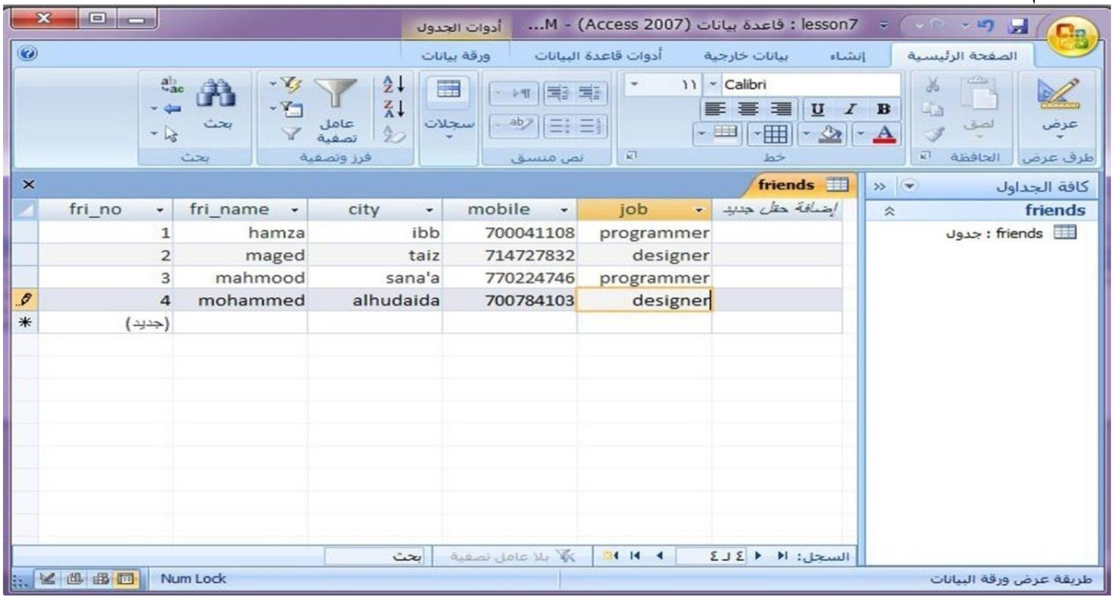


1 / قمنا بتعبئة أسماء الأعمدة في الجدول والنوع البياني لها حسب احتياجنا

واخترنا رقم الصديق أو (fri_no) ليكون مفتاح أساسي
2 / نعود لعرض الجدول في طريقة عرض ورقة البيانات بالضغط على (عرض)
سوف تظهر لنا هل تريد الحفظ ان لم نكن قد حفظنا ماقمنا به بنوافق بالظغط على نعم .
ليظهر لنا الجدول باعمدته خالية من البيانات نقوم بتعبئتها بالبيانات

https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

ونقوم بعدها بالحفظ ويكون شكلها قبل الحفظ هكذا تقريباً



نغلق برنامج الاكسس .

https://www.facebook.com/hamzawey2035

@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

قبل ان ندخل في الربط وطريقته يجب ان نعرف بعض الأمور عن هذه التقنية التي ابتكرتها شركة مايكروسوفت وهي تقنية ado.net وهي التقنية التي نستخدمها في عملية الربط .

هناك ما يسمى بالأعمدة السبعة لتقنية ado.net من مايكروسوفت وهي كلاسات او فئات تقدمها لنا للتحكم بعملية ربط برامج الواجهات مع قواعد البيانات المختلفة وهي

1. فئة الاتصال (connection) .
وهي الفئة المخصصة للاتصال بقواعد البيانات فيحدد مسار قاعدة البيانات وما يتعلق بذلك من وسائط كالتشفير واسم المستخدم وغير ذلك .
2. فئة الأمر (command).
وهي الفئة المخصصة لتنفيذ أمر ما من أوامر SQL المختلفة , أو أي إجراء مخزن في نظام إدارة قاعدة البيانات .
3. فئة قارئ البيانات (Data Reader).
تستخدم للقراءة فقط من قاعدة البيانات في النمط المتصل , والبيانات يمكن قراءتها إلى الأمام ولا يمكن الرجوع إلى الخلف , وذلك بفئة الاستعلام وتعتبر ذات أداء عالي جداً .
4. فئة مهيئ(موصل) البيانات (Data Adapter).
تعمل هذه الفئة على تعبئة البيانات المختلفة من قواعد البيانات إلى مجموعة بيانات معينة Dataset , كما تقوم بتنفيذ التعديلات التي أجريت على مجموعة من البيانات وعكسها على قواعد البيانات .
5. فئة مجموعة البيانات (Dataset).
هي مجموعة من الأنظمة البرمجية تخزن فيها البيانات في الذاكرة وتسمح بإدارتها. أيضاً وهي عبارة عن قاعدة بيانات صغيرة موجودة في الذاكرة تمكننا من انشاء جداول وربط هذه الجداول بعلاقات وهي لاتعتمد على مصدر بيانات محدد.
6. فئة باني الأمر (CommandBuilder).
تستخدم هذه الفئة من قبل الــ Data Adapter لاختيار وإدخال وتحديث وحذف السجلات في قاعدة البيانات , فهذه الفئة تقوم بعمل فئة الأمر ولكن على فئة من نوع مهيئ البيانات .
7. فئة ربط البيانات (Bindingsource).
وهي الفئة المخصصة لربط قيمة إحدى خصائص فئة ما بإحدى خصائص أداة معينة , بحيث يتم وضع قيمة خاصية الأداة تلقائيا كلما تغيرت , والعكس ...

https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

وبإختصار تصبح الخاصيتين كأنهما خاصية واحدة . أي أن استخدام هذه الفئة في ربط محتويات إحدى خصائص الأداة (مثل خاصية النص التابعة لمربع النص) .

❖ هناك شيئا آخر لابد وأن نتحدث عنه فهو مهم ولا تستطيع أن تقول عن نفسك أن بإمكانك استخدام تقنية ado.net بدون معرفة به وهي (providers) وهي وسيطات بين قاعدة البيانات و الاتصال (connection) وهي أنواع تعتمد على نوع قاعدة البيانات المستخدمة في الربط ولن أتعمق فيها كثيراً فهو للمبتدئين وسأذكر بعض أنواعها :

- مزود الـ SQL Server ويكتب بالشكل التالي

Provider=SqlOLEDB;

- مزود الأكسس

أكسس 2003

Provider=Microsoft.Jet.OLEDB.4.0;

أكسس 2007

Provider=Microsoft.ACE.OLEDB.12.0;

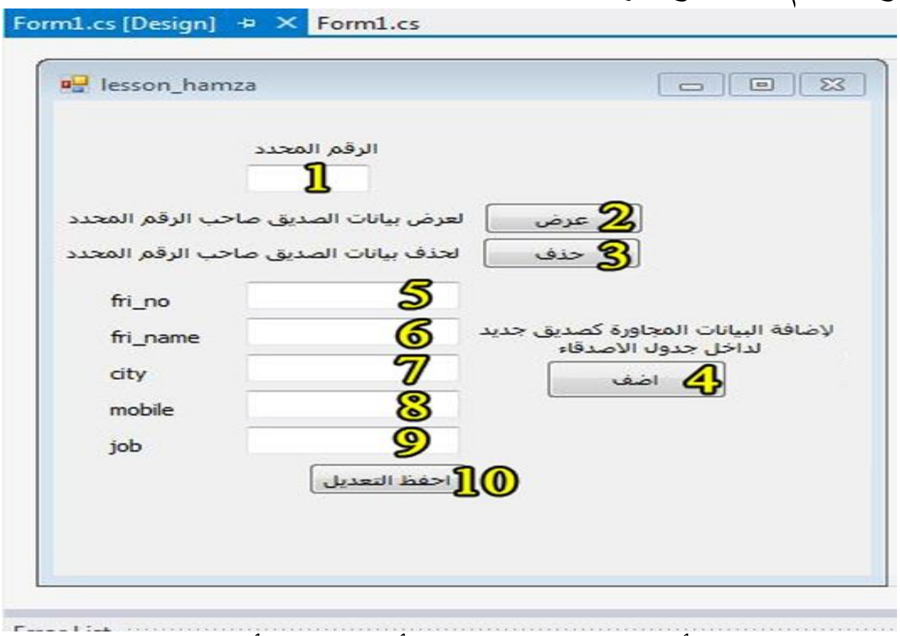
❖ حتى وإن لم تستوعبوا السابق بشكل جيد فإن التطبيق سيوضح لكم الأمر أكثر وسنستخدم بعض من هذه الكلاسات (الفئات) في مشروعنا الأول للربط وسوف ندخل إلى طرق الربط واحدة تلو الأخرى لتترسخ بالذهن فهيا بنا .

https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

- نفتح الفيجوال ستوديو في واجهة البرنامج
نفتح مشروع جديد بلغة السي شارب

File >> new >> project>>visual c# >>windows>>windows form application

ونصمم هذه الواجهة



قمت بترقيم بعض الأدوات المهمة في لأكتب لكم أسماءها لكي نفهم الكود جيداً.

1. صندوق نصي (textbox) وقمنا بتسميته (txt_no) .
وهذا لأجل أن يقوم المستخدم برقم الصديق الذي يريد عرض بياناته في الصناديق النصية .
وأيضاً إن أراد أن يقوم بالتعديل في بيانات احد الأصدقاء في الجدول فهنا يضع رقمه ويقوم بكتابة البيانات الجديدة في الصناديق .
2. زر (button) وقمنا بتسميته (btn_show1) .
هذا الزر لعرض بيانات صديق محدد يكون قد ادخل رقمه في (txt_no).
3. زر (button) وقمنا بتسميته (btn_del) .
هذا الزر لحذف بيانات صديق محدد يكون قد ادخل اسمه في (txt_no).
4. زر (button) وقمنا بتسميته (btn_add) .
هذا الزر يقوم بإضافة صديق جديد للجدول من البيانات المكتوبة في الصناديق الخمسة .
5. صندوق نصي (textbox) وقمنا بتسميته (txt_fri_no) .
رقم الصديق

https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

6. صندوق نصي (textbox) وقمنا بتسميته (txt_fri_name) .
اسم الصديق
7. صندوق نصي (textbox) وقمنا بتسميته (txt_city) .
المدينة
8. صندوق نصي (textbox) وقمنا بتسميته (txt_mobile) .
رقم الجوال
9. صندوق نصي (textbox) وقمنا بتسميته (txt_job) .
الوظيفة
10. زر (button) وقمنا بتسميته (btn_upd) .
يقوم هذا الزر بتعديل بيانات الصديق صاحب الرقم المكتوب في 1 مستخدما بذالك البيانات المكتوبة في الصناديق .

الآن نبدأ

اولاً لنقوم بالربط مابين برنامج وقاعدة بيانات يجب ان نضيف فضاء الاسماء الخاص بهذا النوع من قاعدة البيانات وفضاء الاسماء هو المكاتب وتوجد في نافذة الكود في الأعلى نرى منها غالباً

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
```

وهنا نضيف المكتبة الخاصة بقاعدة البيانات التي سوف نستخدمها بالبرنامج نقوم بتحديد المكتبة التي من نوع قاعدة البيانات التي نشتغل عليها .

- في الـ SQLServer نستخدم المكتبة

```
using System.Data.SqlClient;
```

- في الأكسس نستخدم المكتبة

```
using System.Data.OleDb;
```


https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

الآن سنبدأ بكتابة الأكواد
ندخل إلى نافذة الكود بالضغط على F7 وتحديداً إلى القسم العام (public)
نبدأ بتعريف المتغيرات وكتابة هذي الأكواد

```
    public partial class Form1 : Form
    {
1|        OleDbConnection con = new
OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data Source=lesson7.accdb");
2|        OleDbCommand cmd;
3|        OleDbDataReader dr;
4|        DataTable dt = new DataTable();
        public Form1()
```

ما هي هذه الأكواد ؟

1. قمنا باشتقاق كائن من الكلاس `OleDbConnection` وأسميناه con وهذا الكلاس هو الذي يقوم بتنفيذ الاتصال بقاعدة البيانات ومن الباراميترات الخاصة به انه يستقبل المزود ومسار قاعدة البيانات فلو رأينا الكود نرى أننا أعطيناه أولا المزود الخاص بقاعدة البيانات `Provider=Microsoft.ACE.OLEDB.12.0;` ومن الواضح اننا استخدمنا قاعدة بيانات اكسس 2007 .

ثم اعطيناه مسار قاعدة البيانات واسمها `Data Source=lesson7.accdb` وطبعاً هنا لم نحتاج لكتابة مسار قاعدة البيانات واكتفينا باسمها لأننا وضعناها في مسار البرنامج الأساسي F:\c#\2014\lesson7\lesson7\bin\Debug

2. اشتققنا كائن من فئة `Command` وهي الأوامر وأسميناه `cmd` والذي سنحمله نص الأوامر التي نحتاجها من قاعدة البيانات .
3. اشتققنا كائن من فئة `DataReader` وهي قارئ البيانات واسميناه dr وهو لقراءة البيانات من قاعدة البيانات .
4. اشتققنا كائن من فئة `DataTable` وهي فئة نستطيع أن نحملها قيم جدولية وأسميناه dt .

https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

الآن نبدأ في باقي الأدوات

نعود الى نافذة التصميم بالضغط على SHIFT+F7

1. في الزر رقم 2 والذي قمنا بتسميته (btn_show1) . نكتب الكود التالي

```
2. try
3.             {
4.                 cmd = new OleDbCommand("select * from friends where
   fri_no=" +txt_no.Text +" ", con);
5.                 con.Open();
6.                 dr = cmd.ExecuteReader();
7.                 dr.Read();
8.                 txt_fri_no.Text = dr["fri_no"].ToString();
9.                 txt_fri_name.Text = dr["fri_name"].ToString();
10.                txt_city.Text = dr["city"].ToString();
11.                txt_mobile.Text = dr["mobile"].ToString();
12.                txt_job.Text = dr["job"].ToString();
13.                dr.Close();
14.
15.                MessageBox.Show("العرض تم ");
16.            }
17.            catch (OleDbException xx)
18.            {
19.                MessageBox.Show("ما خطأ هناك " + xx.Message);
20.            }
21.            finally
22.            {
23.                con.Close();
            }
```

المهم في هذا الكود هو الأسطر من 4 الى 13 واستخدمنا دوال try و catch لتفادي الوقوع في اخطاء فقط تعالوا نشرح الكود .

4/ حملنا الكائن cmd نص الأمر و مفتاح الأتصال ونص الأمر اشترطنا فيه أن يكون رقم الصديق يساوي النص المدخل في (txt_no)

5/ فتحنا الاتصال باستخدام دالة Open(); التابعة للكائن con .

6/ أسندنا البيانات التي قام بطلبها cmd لقارئ البيانات dr

7/ جعل dr يقرأ البيانات .

8/أدخلنا في الخاصية Text للصندوق النصوص (txt_fri_no) قيمة قمنا بأخذها من dr وقمنا بتحديد الحقل الذي سنأخذ منه البيانات الى هذا الصندوق بالأمر ["fri_no"].ToString(); حيث ["fri_no"] هي اسم الحقل في الجدول سنقوم بعرضها في صندوق النصوص هذا ونستطيع ان نكتب رتبة الحقل مثلاً هنا نستطيع أن نكتب 0 ولكن لتفادي الوقوع بأخطاء نكتب اسم الحقل .

https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

9/ وكما فعلنا مع صندوق النص الخاص برقم الصديق نفعل مع بقية الصناديق محددين كل صندوق نصي ما هو الحقل الذي سيعرضه .
13/ نقوم بإغلاق الاتصال.

Try & Catch & Finally

جملة try بالمعنى هي جرب أو حاول وهي إخبار المترجم أن يجرب أن ينفذ الكود الموجود بين الحاصرتين التابعة لها لتفادي الأخطاء ويجب بعد استخدام try يجب استخدام دالة catch والخلاصة أن try لها حاصرة بداية وحاصرة نهاية وما بينهما يكتب الكود الذي تريد تنفيذه ثم بعد قفل الحاصرة يجب استخدام catch وهي تقوم بمقام الخيار الثاني الذي ينتقل اليه المترجم إذا كان هناك خطأ بالكود الذي في داخل try والـ catch ايضا لها حاصرتين يكتب في داخلها الكود المراد تنفيذ إذا لم يتم تنفيذ جملة try ونستطيع أيضا قبل فتح حاصرة ال catch اشتقاق كائن من أي كلاس لنستخدمه وفي هذا المثال قمنا باشتقاق كائن من الفئة OleDbException وهو بمقام الاستثناء بلغة الأس كيو إل واسمينا الكائن xx وجعلنا في جملة catch رسالة تخبرنا بالخطأ الذي وقع واستخدمنا دالة في الكائن xx لتعرض لنا نص الرسالة . عرفنا هنا ان جملة ال catch تنفذ اذا لم تنفذ جملة الـ try أما الـ finally فهي دالة تنفذ جملها في كلتا الحالتين سواء تنفذت جمل try او جمل catch .

الان ننتقل الى الزر الثاني

في الزر رقم 3 والذي قمنا بتسميته (btn_del) .
نقوم بكتابة هذه الأكواد

```
cmd = new OleDbCommand("delete from friends where fri_no=" + txt_no.Text + "",
con);
            con.Open();
            cmd.ExecuteNonQuery();
                MessageBox.Show("تم حذف بيانات الصديق رقم   " + txt_no.Text);
```

في السطر الأول قمنا بأعطاء كائن الأوامر الأمر اللازم بحذف بيانات الصديق صاحب الرقم المحدد في ال صندوق النص (txt_no)
مع مفتاح الأتصال
ثم فتحنا الأتصال بالسطر الذي يليه
ثم قمنا بتنفيذ الأمر بإستخدام الدالة ;()ExecuteNonQuery
و هذه الدالة نستخدمها لتنفيذ الأمر مثلها مثل ;()ExecuteReader
والفرق بينهما أن الأولى نستخدمها إذا كان هناك بيانات مرتجعة من الأمر أما الثانية فنحن نستخدمها عندما لا يكون هناك بيانات مرتجعة مثل عمليات الإضافة والحذف والتعديل .

https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

وفي السطر الذي يليه قمنا بإظهار رسالة للمستخدم تفيد بأنه تم الحذف ويمكننا بعده إغلاق الاتصال ولم اكتب كود إغلاق الاتصال لأني قمت باستخدام دوال التفادي من الأخطاء ووضعت كود الإغلاق فيها .

ننتقل للزر الذي يلية

ننتقل للزر الذي يليه وهوالزر رقم 4 والذي قمنا بتسميته (btn_add) ونكتب به هذا الكود

```
try
            {
(1)                 cmd = new OleDbCommand("insert into friends
(fri_no,fri_name,city,mobile,job) values ('" + txt_fri_no.Text + "','" +
txt_fri_name.Text + "','" + txt_city.Text + "','" + txt_mobile.Text + "','" +
txt_job.Text + "')", con);
(2)                 con.Open();
(3)                 cmd.ExecuteNonQuery();
(4)                 MessageBox.Show("الاضافة تمت ");
            }
            catch (OleDbException xx)
            {
                MessageBox.Show(" خطأ    " + xx.Message);
            }
            finally
            {
                con.Close();
            }
```

الآن للشرح

سأقوم بشرح الأسطر الذي قمت بترقيمها باللون الأخضر من 1 : 4 فلا داعي لأن اشرح دوال **Try & Catch & Finally** فأعتقد إنكم تعرفونها فقد تحدثنا عنها سابقاً

1. كما قمنا سابقاً بإعطاء الكائن cmd أمر الـ sql و مفتاح الاتصال الذي هو في مشروعنا con وهكذا فعلنا في هذا السطر قمنا باستخدام أمر الإضافة الخاص بالـ sql ولكي نأخذ القيم (values) من داخل الصناديق النصية نستخدم العلامة + الذي نستخدمها في دمج السلاسل النصية او الربط بينها ولكن يجب التركيز على التنصيصات الأحادية والمزدوجة وبعد الأمر قمنا بتسمية كائن الاتصال والذي هو con .
2. فتحنا الاتصال ;()con.Open .
3. قمنا بتنفيذ الأمر باستخدام الدالة ;()cmd.ExecuteNonQuery .
4. قمنا بعرض رسالة توضيحية للمستخدم تفيد بأنه قد تمت الإضافة .

لاحظوا أننا لم نكتب كود إغلاق الاتصال ؟؟ ولكنا قمنا بكتابة في دالة finally الذي ستنفذ في جميع الأحوال .

https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

الآن ننتقل إلى الزر التالي
وهو الزر رقم 10 والذي قمنا بتسميته (btn_upd) . والذي يقوم بحفظ التعديل
ونكتب فيه هذا الكود

```
try
            {
                cmd = new OleDbCommand("update friends set fri_no=" +
txt_fri_no.Text + ", fri_name='" + txt_fri_name.Text + "',city='" +
txt_city.Text + "',mobile=" + txt_mobile.Text + ",job='" + txt_job.Text + "'
where fri_no=" + txt_no.Text + "", con);
                con.Open();
                cmd.ExecuteNonQuery();
                MessageBox.Show("تم التعديل ");
            }
            catch (OleDbException xx)
            {
                MessageBox.Show(" خطأ    " + xx.Message);
            }
            finally
            {
                con.Close();
            }
```

عندما نرى الكود نرى انه قد تكرر لدينا اكثر من مره وهو مشابة بدرجة كبيرة لكود الاضافة والاختلاف الوحيد هو كود الـ sql وهو يقوم بوضع القيم الجديدة من الصناديق وفق رقم الصديق المكتوب بالصندوق الأول .

https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

ما هي البيئة المتصلة و البيئة المنفصلة ؟؟؟

الدرس السابق كان في إطار البيئة المتصلة و هي التي نكون فيها على اتصال دائم بقاعدة البيانات

إما البيئة المنفصلة فهي عكس ذلك أي ليست على اتصال دائم بقاعدة البيانات و إنما تأخذ نسخة من قاعدة البيانات وتشتغل بها إلى ان تتم الحاجة للاتصال

ما الفائدة من البيئتين ولماذا صممت مايكروسوفت الأداتين ؟؟

إن مايكروسوفت عندما أضافت هذه التقنية و استطعنا أن نتعامل مع قواعد البيانات في الوضع المنفصل خدمة كبيرة جداً سأشرح لك لماذا وما الفائدة من جعل هناك بيئتين .

إننا نختار البيئة المتصلة عندما نكون نتعامل مع سرفر محلي وقاعدة بيانات واحدة ويستخدمها مستخدم واحد و هنا يكون الوضع المتصل أفضل للاتصال لكن إن كان هناك شبكات تتاح لأكثر من مستخدم على نفس قاعدة البيانات أي انه في الوضع المنفصل كل التعديلات الذي تقوم بها على قاعدة البيانات تحفظ على جدول و همي مؤقت داخل برنامجك وليس قاعدة البيانات و عند الوقت اللازم يتم حفظها إلى قاعدة البيانات

https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

البيئة المنفصلة

قبل ان ندخل في تطبيقات البيئة المنفصلة سنقوم بجولة بسيطة لنتعرف إلى بعض المفاهيم الذي سنحتاجها سواء كنا نتعامل في البيئة المتصلة او المنفصلة وسأذكر بعض المفاهيم التي تدخل في تقنية الـ ado.net .

- مجموعة البيانات (Data set)

هي مجموعة من الأنظمة البرمجية تخزن فيها البيانات في الذاكرة وتسمح بإدارتها .أيضا هي عبارة عن قاعدة بيانات صغيرة موجودة في الذاكرة تمكننا من إنشاء جداول وربط هذه الجداول بعلاقات وهي لا تعتمد على مصدر بيانات محدد .

ومن بعض مكونات الـ Data set

- ✓ **الفئة جدول البيانات Data Table :**
عبارة عن تمثيل لجدول إما أن تقوم ببناء هيكله في Dataset أو تأخذه من مصدر بيانات ويمكن أن تحتوي Dataset على عدد غير محدد من الجداول .
- ✓ **الفئة أعمدة البيانات Data Column :**
عبارة عن تمثيل لعمود داخل جدول هذا العمود له نوع واسم وطول الخ ... من مواصفات الأعمدة.
- ✓ **الفئة صفوف البيانات Data Row :**
يمثل هذا الفئة سجل في الفئة Data Table حيث تؤمن طريقته Item الوصول الى قيم حقوله .
- ✓ **الفئة قيود الأعمدة Data Constariat :**
قيود توضع على الأعمدة (تتحكم بهيئة البيانات داخل الأعمدة) مثل
(Primary , Foreign , NULL , Constariat , Unique)
- ✓ **الفئة علاقات DataRelations :**
نستطيع من خلاله إنشاء (تحديد) العلاقات بين الجداول .
- ✓ **الفئة مستعرضات Data View :**
إنشاء مستعرض يحتوي على جدول أو أكثر من خلال هذا الكائن .

وسنذكر هنا بعض الخصائص والكائنات الذي سنحتاجها سواء في درسنا هذا أو فيما بعد :

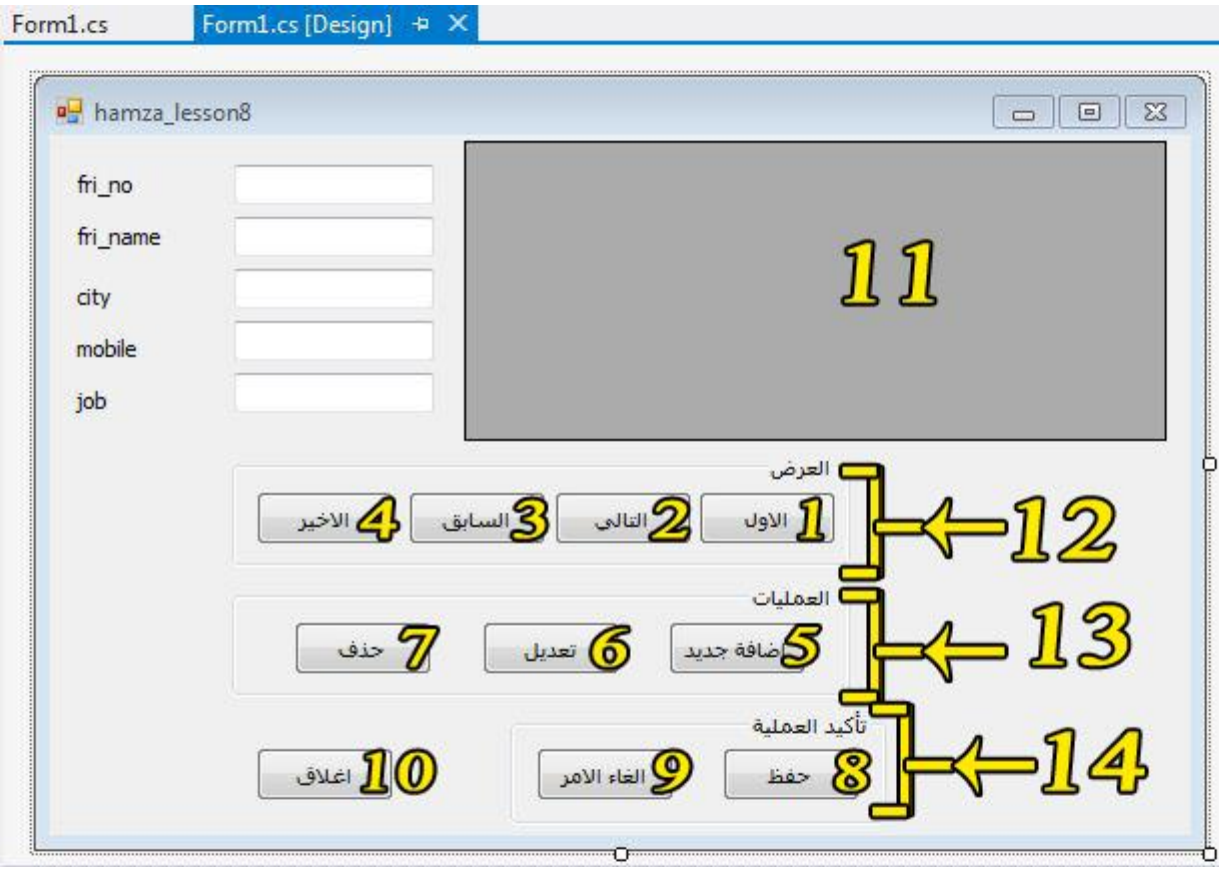
- ❖ **الخاصية Position :**
نستخدمها للتنقل بين السجلات وهي تتبع المتغير الذي تم تعريفة من نوع BindingManagerBase .
- ❖ **الخاصية DataBinding :**
تستخدم هذه الخاصية لربط كائن مع عمود في جدول وعرض محتواه فيه .
- ❖ **الخاصية BindingContex :**
يستخدم لربط الكائنات المرتبطة مع الـDataset والجدول والتحكم بسجلات الجدول عن طريق استخدام متغير من نوع BindingManagerBase .
- ❖ **BindingManagerBase :**
وهو كائن يقوم بتنسيق العلاقة بين كائنات الربط ومجموعة البيانات ويقوم بالتحكم بالحركة داخل السجلات .
- ❖ **CurrencyManager :**
وهو كائن يقوم بنفس عمل BindingManagerBase .
- ❖ **الخاصية displayMember التابعة لـ ComboBox :**
تستخدم في ربط قاعدة البيانات بحقل من الجدول .

https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

- **الخاصية ValueMember التابعة لـ ComboBox :**
وهي قيمة DisplayMember التي سوف يتم إرجاعها إلى قاعدة البيانات .
- **الخاصية DataSource التابعة لـ ComboBox :**
وهي طريقة تقوم بتحديد اسم قاعدة البيانات .
- **كائن باني الأمر CommandBuilder :**
وهو كائن يستخدم من قبل الـ DataAdepter لاختيار وإدخال وتحديث وحذف السجلات في قاعدة البيانات .
- **الخاصية InsertCommand التابعة لـ DataAdapter :**
وهي خاصية بالفئة DataAdapter وتعتبر كمرجع للفئة CommandBuilder والتي يستخدمها الـ DataAdapter لإضافة البيانات بين قاعدة البيانات وفئة الـ Dataset .
- **الخاصية DeleteCommand التابعة لـ DataAdapter :**
وهي خاصية بالفئة DataAdapter وتعتبر كمرجع للفئة CommandBuilder والتي يستخدمها الـ DataAdapter لحذف البيانات بين قاعدة البيانات وفئة الـ Dataset .
- **الخاصية UpdateCommand التابعة لـ DataAdapter :**
وهي خاصية بالفئة DataAdapter وتعتبر كمرجع للفئة CommandBuilder والتي يستخدمها الـ DataAdapter لتعديل البيانات بين قاعدة البيانات وفئة الـ Dataset .
- **الخاصية SelectCommand التابعة لـ DataAdapter :**
وهي خاصية بالفئة DataAdapter وتعتبر كمرجع للفئة CommandBuilder والتي يستخدمها الـ DataAdapter لاستعلام أو اختيار البيانات بين قاعدة البيانات وفئة الـ Dataset .
- **الوظيفة GetInsertCommand التابعة لباني الأمر CommandBuilder :**
وهي عبارة عن خاصية للفئة CommandBuilder تقوم بإضافة السجلات الجديدة من الـ Dataset وإسقاطها على الـ DataAdapter عن طريقة الخاصية (مرجع) InsertCommand .
- **الوظيفة GetDeleteCommand التابعة لباني الأمر CommandBuilder :**
وهي عبارة عن خاصية للفئة CommandBuilder تقوم بحذف السجلات المحذوفة من الـ Dataset وإسقاطها على الـ DataAdapter عن طريقة الخاصية (مرجع) DeleteCommand .
- **الوظيفة GetUpdateCommand التابعة لباني الأمر CommandBuilder :**
وهي عبارة عن خاصية للفئة CommandBuilder تقوم بتعديل السجلات التي تم تعديلها من الـ Dataset وإسقاطها على الـ DataAdapter عن طريقة الخاصية (مرجع) UpdateCommand .

اعتقد أن الأمور واضحة وحتى إن لم تستوعبوا السابق بشكل جيد فبالتطبيق سيزيل الالتباس والغموض وتصبح الأمور جلية أكثر هيا بنا للتطبيق نعود مرة أخرى إلى برنامج الفيجوال ستوديو ننشئ مشروع جديد ونصمم هذه الواجهة :

https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com



شرح الواجهة :

- ❖ الصناديق النصية على جهة الشمال هي كما في التطبيق السابق .
- ❖ في جهة اليمين هي أداة dataGridView أداة لعرض البيانات وأسميناها dataGridView1 وهي الظاهرة معنا بالصورة بالرقم 11.
- ❖ مجموعة الأزرار الأولى

هي الأدوات المرقمة من (1-4) هي أزرار (button) وهي للقيام بعمليات التنقل بين بيانات قاعدة البيانات وأيضاً نحن وضعناها هذه الأزرار داخل أداة (groupBox) وأسميناها groupBox1 وهي الظاهرة معنا في الصورة بالرقم 12.

- ❖ مجموعة الأزرار الثانية

هي الأدوات المرقمة من (5-7) هي أزرار (button) وهي للقيام بعمليات الإضافة والتعديل والحذف لبيانات قاعدة البيانات وأيضاً نحن وضعناها هذه الأزرار داخل أداة (groupBox) وأسميناها groupBox2 وهي الظاهرة معنا في الصورة بالرقم 13.

- ❖ مجموعة الأزرار الثالثة

هي الأداتين التي أرقامها(8 و 9)هي أزرار(button)وهي للقيام بتأكيدعمليات الإضافة والتعديل والحذف لبيانات قاعدة البيانات وأيضاً نحن وضعناها هذه الأزرار داخل أداة (groupBox) وأسميناها groupBox3 وهي الظاهرة معنا في الصورة بالرقم 14.

- ❖ الزر رقم (10) وهو الزر الذي يقوم بإغلاق البرنامج .

https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

س ما أول ما نقوم به ؟

لا شك أن أول ما يجب علينا القيام به هو تعريف المكتبة التي نحتاجها وهنا نحتاج المكتبة التي تخص قاعدة البيانات أكسس والتي هي

```
using System.Data.OleDb;
```

ونبدأ تعريف المتغيرات واشتقاق الكائنات في القسم العام ونكتب التالي

```
public partial class Form1 : Form
    {
        // نقوم بتعريف جملة الاتصال  كما سبق ذكره قبلاً
        OleDbConnection con = new
OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data
Source=lesson7.accdb");
        // تعريف كائن من فئة  منظم البيانات
        OleDbDataAdapter da;
        // تعريف كائن من فئة جدول البيانات
        DataTable dt = new DataTable();
        // تعريف كائن من فئة العلاقات
        BindingManagerBase bmb;
        // تعريف كائن من فئة باني الأوامر
        OleDbCommandBuilder cmdb;
        public Form1()
        {
            InitializeComponent();
        }
```

سأسألكم سؤالاً لماذا وضعنا بعض الأدوات داخل الحاويات (groupBox) ؟؟

للتنظيم صحيح ! أجل يمكن ولكن الغرض الذي وضعناها لأجله هو أننا نريد عند اشتغال البرنامج فإن الأزرار رقم (8 و 9) الذي يقومان بتأكيد العمليات لا نريد ان يظهرا على المستخدم الا عند اختيار المستخدم إحدى العمليات (الإضافة – التعديل – الحذف) وتختفي بقية الأدوات وعند الضغط على أحد الزرين يعود الوضع للحالة السابقة ولهذا سنجعل الخاصية Enabled للأداة groupBox3 = False ونكتب هذه الدالة داخل البرنامج والتي سنستخدمها عند الحاجة :

```
void toolsShow()
        {
            groupBox1.Enabled = !groupBox1.Enabled;
            groupBox2.Enabled = !groupBox2.Enabled;
            groupBox3.Enabled = !groupBox3.Enabled;
        }
```

وهذه الدالة عند استدعاءها تقوم بقلب حالة تشغيل الحاويات على ما هي عليه .

https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

نعود إلى واجهة تصميم الواجهة بالضغط على SHIFT+F7 ونضغط على الفورم مرتين في حدث اللود نكتب الكود التالي :

```
private void Form1_Load(object sender, EventArgs e)
        {
            // إعطاء كائن منظم البيانات جملة الإس كيو إل وإيضاً مفتاح الإتصال
            da = new OleDbDataAdapter("select * from friends", con);
            // استخدام الدالة فيل من دوال كائن منظم البيانات لتعبئة متغير جدول البيانات
            // أي أنه بعد تنفيذ الكود السابق يكون منظم البيانات كأنه يحمل
            // البيانات الراجعة من قاعدة البيانات ونطلب منها هنا تعبئتها للجدول ويمكننا فعل التالي بالأمر
            //dt.load(da);
            // أي أننا نجعل كائن جدول البيانات يقوم بتحميل البيانات من منظم البيانات
            da.Fill(dt);
             // وهنا نقوم بربط خاصية من خواص الأداة بحقل معين من حقول الجدول
             //بإستخدام خاصية DataBindings وتستقبل بعض القيم كبارامترات
             // البارامبتر الأول خاصية الأداة التي سنضيف اليها بيانات الحق و هنا استخدمناtext
             // والثاني اسم الجدول او مصدر البيانات والذي هو عندناdt
             // والثالث هو اسم العمود في الجدول المحدد والذي هو عندناfri_no
وهذا يعني أننا قمنا بربط خاصية النص لصندوق النص هذا بالحقل المحدد في العمود رقم الصديق بالجدول
        // dt
            txtfri_no.DataBindings.Add("text", dt, "fri_no");
             //كما سبق إلا اننا هنا ربطناه بإسم الصديق
            txtfri_name.DataBindings.Add("text", dt, "fri_name");
             //وهذا ربطناه بالمدينة
            txtcity.DataBindings.Add("text", dt, "city");
             //وهذا برقم الجوال
            txtmobile.DataBindings.Add("text", dt, "mobile");
             //وهذا بالوظيفة
            txtjob.DataBindings.Add("text", dt, "job");
             // قمنا بالربط بين الـ  bmb و الـdt
             // او نستطيع القول اننا حددنا لـ bmb الكائن الذي عليها التعامل معه
            bmb = this.BindingContext[dt];
             // قمنا بتعبئة أداة عرض البيانات ببيانات الجدولdt
            dataGridView1.DataSource = dt;
        }
```

يمكنك ان تجرب تشغيل البرنامج لترى انه قد عرض لنا بيانات الجدول !!!

و الآن نرجع إلى نافذة التصميم على الزر الأول والذي يعرض بيانات الصديق الأول ونكتب فيه هذا الكود

```
private void btnfrist_Click(object sender, EventArgs e)
        {
            // جعلنا موقع الـ bmb = 0 أي اننا نخبره عن بيانات الصديق الذي يحتل الرتبة صفر
            bmb.Position = 0;
        }
```


https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

ونرجع إلى واجهة التصميم ونضغط على الزر الثاني والذي يعرض بيانات الصديق التالي ونكتب هذا الكود

```
private void next_Click(object sender, EventArgs e)
        {
            // زدنا موقع الـ bmb بنسبة 1 أي اننا نخبره عن بيانات الصديق الذي يلي الصديق المعروض حالياً
            bmb.Position++;
        }
```

ونرجع إلى واجهة التصميم ونضغط على الزر الثالث والذي يعرض بيانات الصديق السابق ونكتب الكود التالي

```
private void btnprevious_Click(object sender, EventArgs e)
        {
            // انقصنا موقع الـ bmb بنسبة 1 أي اننا نخبره عن بيانات الصديق الذي يسبق الصديق المعروض حالياً
            bmb.Position--;
        }
```

نرجع إلى واجهة التصميم نضغط على الزر الرابع والذي يعرض بيانات الصديق الأخير ونكتب الكود التالي

```
private void btnlast_Click(object sender, EventArgs e)
        {
            // جعلنا موقع الـ bmb = اقل من عدد صفوف الجدول بمقدار أ
            bmb.Position = dt.Columns.Count - 1;
        }
```

وفي الزر رقم 5 والذي يقوم بعملية الإضافة نقوم بكتابة الكود التالي

```
private void btnAdd_Click(object sender, EventArgs e)
        {
            // قمنا بإخبار المعالج اننا نريد اضافة بيانات جديدة
            bmb.AddNew();
            //استدعينا الدالة التي قمنا بكتابتها لتغيير حالة تشغيل الأزرار
            toolsShow();
        }
```

وفي الزر رقم 6 والذي يقوم بعملية التعديل لا نكتب إلا كود استدعاء دالة تغيير حالة تشغيل الأزرار

وفي الزر رقم 7 والذي يقوم بعملية الحذف نكتب الكود التالي :

```
private void btnDel_Click(object sender, EventArgs e)
        {
            // وهذا السطر يقوم بعرض رسالة للمستخدم تقول له هل انت متأكد من عملية الحذف
            // فإن اختار نعم تحقق شرط وتم تنفيذ ماداخل الحاصرتين او بالأصح تم تنفيذ جملةIf
            //وان اختار لا تم الخروج من الجملة وعدم حدوث اي تغيير
            if (MessageBox.Show("هل انت متأكد من عملية الحذف ", "الحذف تأكيد ",
MessageBoxButtons.YesNo) == DialogResult.Yes)
            {
                //إعطاء أمر الحذف بالدالة Removeat والتي تقوم بالحذف حسب الموقع وتستقبل الموقع كبارامتر
                bmb.RemoveAt(bmb.Position);
                //نعطيه الأمر بإنهاء العملية الحالية
                bmb.EndCurrentEdit();
                // حددنا لباني الأوامر منظم البيانات الذي عليه التعامل معه
                cmdb = new OleDbCommandBuilder(da);
                // منظم البيانات يقوم بتحديث البيانات الذي في الـdt
                da.Update(dt);
            }
        }
```


https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

ننتقل للزر رقم 8 والذي يقوم بحفظ أو تأكيد العملية ونكتب هذا الكود

```
private void btnSave_Click(object sender, EventArgs e)
        {
            //نعطيه الأمر بإنهاء العملية الحالية
            bmb.EndCurrentEdit();
            //  حددنا لباني الأوامر منظم البيانات الذي عليه التعامل معه
            cmdb = new OleDbCommandBuilder(da);
            //  منظم البيانات يقوم بتحديث البيانات الذي في الـdt
            da.Update(dt);
            //استدعينا الدالة التي قمنا بكتابتها لتغيير حالة تشغيل الأزرار
            toolsShow();
        }
```

ننتقل للزر رقم 9 والذي يقوم بإلغاء العملية ونكتب الكود التالي

```
private void btnCancel_Click(object sender, EventArgs e)
        {
            // CancelCurrentEdit(); نعطيه الأمر بإلغاء العملية بإستخدام الدالة
            bmb.CancelCurrentEdit();
            //استدعينا الدالة التي قمنا بكتابتها لتغيير حالة تشغيل الأزرار
            toolsShow();
        }
```

أما في زر الإغلاق فهو أمر إغلاق البرنامج فقط كما تعرفون

```
private void btnClose_Click(object sender, EventArgs e)
        {
            this.Close();
        }
```


https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

❖ المثال السابق هو يوضح الربط مابين بيانات قاعدة البيانات وخواص الأدوات في برنامجك في الوضع الغير متصل اعتقد أن الأمر جلياً أننا في البيئة المنفصلة نستخدم الداتا ادابتر وهو الذي يجعلنا غير متصلين بقاعدة البيانات . المهم أريد أن أضيف لكم مثالاً ضرورياً فيدو أن العديد منكم قد تساءل لم لا نجعل طريقة اختيار المستخدم للصديق المراد عرض بياناته في قائمة الليست بوكس أو الكومبو بوكس ولهذا التساؤل بل الحاجة الملحة لتعلم ذلك هي ما جعلتني أصر أن أمر بكم على هذا الموضوع بلمحة بسيطة لتأخذوا فكرة عن هذا الموضوع وتفهموا أساسها وبعدئذ تطوروها بأنفسكم لا أطيل عليكم هيا بنا إلى المثال .

❖ نريد إن نصمم برنامج فيه كومبو بوكس يعرض لنا أسماء الأصدقاء وعند اختيارنا لأحد الأصدقاء يعرض بياناته في صناديق نصية .

نصمم هذه الواجهة في برنامج الفيجوال ستوديو



نذهب الى الكومبو بوكس وفي الخاصية DropDownStyle نجعلها DropDownList وذلك لجعلها للقراءة فقط أو بالأصح لمنع التعديل على البيانات المعروضة عليها .
ثم ندخل إلى نافذة الكود وبعد كتابة فضاء الأسماء الخاص بقاعدة البيانات ننزل إلى تعريف المتغيرات واشتقاق الكائنات في الجزء العام ونكتب الكود التالي :

```
        // نقوم بتعريف جملة الاتصال كما سبق ذكره قبلاً
        OleDbConnection con = new
OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data Source=lesson7.accdb");
        // إشتقاق كائن فئة منظم أو جالب البيانات
        OleDbDataAdapter da;
        // تعريف كائن من فئة جدول البيانات ليستقبل البيانات الراجعة
        // وهذا الذي سنستخدمه في تعبئة الكومبو بوكس
        DataTable dt = new DataTable();
        // اشتقاق كائن آخر من فئة جدول البيانات ليستقبل البيانات الراجعة
         // وهذا الذي سنستخدمه في تعبئة الصناديق النصية
        // لأن هذا الجدول سيحمل بيانات سطر واحد أما السابق فهو يحمل بيانات عمود
        DataTable dt2 = new DataTable();
        public Form1()
        {
            InitializeComponent();
        }
```

أظن أن الشرح واضحاً الآن نعود إلى نافذة التصميم ونضغط على الفورم مرتين لننتقل إلى حدث اللود أي حدث تحميل الفورم ونكتب الكود التالي :

```
private void Form1_Load(object sender, EventArgs e)
        {
            // إعطاء كائن منظم البيانات أو جالبها  جملة الإس كيو إل وإيضاً مفتاح الإتصال
            da = new OleDbDataAdapter("select * from friends", con);
            /*
             استخدام الدالة فيل من دوال كائن منظم البيانات لتعبئة متغير جدول البيانات
             أي أنه بعد تنفيذ الكود السابق يكون منظم البيانات كأنه يحمل
             البيانات الراجعة من قاعدة البيانات ونطلب منها هنا تعبئتها للجدول ويمكننا فعل التالي بالأمر
             dt.load(da);
             أي أننا نجعل كائن جدول البيانات يقوم بتحميل البيانات من جالب البيانات
            */
            da.Fill(dt);
            // ثم نحدد مصدر بيانات الكومبو بوكس أنها من الجدول
            comboBox1.DataSource = dt;
            // نحدد الحقل الذي نريد إظهاره في الكومبو بوكس
            // وهنا حددنا الحقل رقم الصديق بأستخدام خاصية DisplayMember للكومبو بوكس
            comboBox1.DisplayMember = "fri_name";
          نكتب اسم الحقل صاحب المفتاح الرئيسي في جدول الأصدقاء في الخاصية ValueMember للكومبو بوكس
 //
            comboBox1.ValueMember = "fri_no";
        }
```

الآن نحن قد حملنا الكومبو بوكس ببيانات العمود اسم الصديق لجميع الأصدقاء ! جرب التنفيذ وسترى.

نعود إلى نافذة التصميم وننقر على الكومبو بوكس نقرتين لندخل الى حدث SelectedIndexChanged لها ونكتب الكود التالي :

```
private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
        {
            try
            {
                dt2.Clear();
                da = new OleDbDataAdapter("select * from friends where fri_name='"
+comboBox1.Text + "'", con);
                // لاشك اننا نعرف ان الإستعلام السابق يرجع لنا سطر من البيانات
                // وهنا قمنا بتعبئة الـ dt2 ببيانات هذا السطر
                da.Fill(dt2);
                /*
                  الآن نأتي لأظهار بيانات السطر الذي في الجدول في الصناديق النصية
              نحدد الجدول ثم نستخدم دالة Rows وهي دالة تحدد الصف وتستقبل رتبة السطر
                  ورتبة السطر عندنا هو الصفر لإن الإستعلام السابق يجلب سطر واحد فقط
                  وتستقبل إيضاً رتبة الحقل ورتبة الحقل في رقم الصديق هي 1
                  ولكن لتفادي الأخطاء نكتب اسم الحقل أفضل من رتبته
                  وعلى نفس الطريقة نكمل باقي الصناديق النصية
                 */
                txt_fri_no.Text = dt2.Rows[0]["fri_no"].ToString();
                txt_fri_name.Text = dt2.Rows[0]["fri_name"].ToString();
                txt_city.Text = dt2.Rows[0]["city"].ToString();
                txt_mobile.Text = dt2.Rows[0]["mobile"].ToString();
                txt_job.Text = dt2.Rows[0]["job"].ToString();
            }
            catch
            {
                return;
            }
        }
```


https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

ونستطيع أيضاً إضافة بيانات للكومبو بوكس او الليست بوكس بعدة طرق منها بالبيئة المتصل باستخدام الداتا ريدر بهذه الطريقة

```
private void button1_Click_1(object sender, EventArgs e)
        {
            try
            {

                cmd = new OleDbCommand("select * from friends", con);
                con.Open();
                dr = cmd.ExecuteReader();
                comboBox1.DisplayMember = "fri_name";
                comboBox1.ValueMember = "fri_no";
                while (dr.Read())
                {
                    comboBox1.Items.Add(dr["fri_name"].ToString());
                }
                dr.Close();

                MessageBox.Show("تم الاضافة ");
            }
            catch (OleDbException xx)
            {
                MessageBox.Show("هناك خطأ ما " + xx.Message);
            }
            finally
            {
                con.Close();
            }
        }
```


https://www.facebook.com/hamzawey2035
@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com

- ❖ يجب أن أنبهكم أننا في الدروس السابقة كلها كنا نضع قاعدة البيانات داخل مجلد البرنامج وان لم تكن هناك يجب علينا كتابة موقع قاعدة البيانات ولوضعها في مسار البرنامج كما كنا نفعل طيلة الدروس السابقة نفتح مجلد البرنامج (bin) ومن داخلة ندخل مجلد (Debug) ونضعها هنا لكي تكون في مسار البرنامج .
- ❖ وإن تريدون أن لا تكتبوها انتم تستطيعون من واجهة الفيجوال الضغط على تبويب (Data) واختيار (Add New Data Source) ثم الضغط على زر (Next) ثم نضغط (New Connection) ثم من زر (Browser) ثم نحدد مكان قاعدة البيانات التي نريد استخدامها ونضغط (Open) لديك في النافذة زر مكتوب عليه (Test Connection) لاختبار نجاح الاتصال ثم الضغط على (OK) فتفتح لنا الواجهة السابقة التي تحمل اسم (Data Source Configuration Wizard) نرى في الأسفل أداة تفرع عليها علامة + ومكتوب عليها (Connection String) نضغط على علامة + فيظهر لنا أسفلها كود الاتصال نستطيع من هنا أن نقوم بنسخة ولصقه في مكان تحديد نص الاتصال .

أعتذر إذا قصرت في بعض الأشياء ولكني قد قلت لكم في البداية هذا الكتاب هو مقدمة فقط ولكنه يحوي أهم الأشياء في هذه التقنية وتقبلوا خالص تحياتي لأي استفسار يرجى مراسلتي على احد الإيميلات وانتظرونا في الكتب القادمة .

X

حمزه علي البعسي
برمجة حاسوب - اليمن

https://www.facebook.com/hamzawey2035

@ alzeerhamza@gmail.com
@ hamzaweynet@gmail.com